DUTIR at TREC 2008 Blog Track

Jianmei Chen, Wei Guo, Fengming Pan, Fuyang Chang, Rui Song, Hongfei Lin
Information Retrieval Lab, Dalian University of Technology, China,
irlab@dlut.edu.cn

ABSTRACT

For opinion finding task our method of the combination of 5 Windows method and Pseudo Relevance Feedback behaves well, achieving an improvement of over 20% on the baseline adhoc results. For the polarity task we develop two different methods. One is a classification method, and the other uses queries to retrieve positive and negative documents respectively. In Blog Distillation task, Pseudo Relevance Feedback method helps improve the result a little, however, since its dependence on the top 10 retrieval result, the method still need to be improved in order to get better result.

1. INTRODUCTION

IR Lab (Information Retrieval Laboratory) of DUT (Dalian University of Technology) participated in all of the tasks of the TREC 2008 Blog Track, including Baseline adhoc(blog post)retrieval task, Opinion Retrieval Task, Polarity Task and Feed Distillation Task. The paper gives a detailed description of the participation. Modules designed for different tasks are described separately below.

2. BASELINE ADHOC TASK

The baseline adhoc retrieval task requires locating blog posts that contain relevant information about a given topic target, however all opinion-finding retrieval techniques should be turned off.

Since the corpus was cleared last year, it was indexed by indri directly. Then we constructed queries in two ways. The first way is just using the title field in the topics. The second way is using some query expansion when the result number was not attained to the requirement. We expanded queries by Wiki, indri query syntaxes. In a word, the methods we used in this part were very simple and we haven't used any opinion-finding retrieval techniques. The results are as follow and can be used as baselines for the next parts.

RunID	Query Description	MAP (topicrel)	R-Precision (topicrel)	P@10 (topicrel)	MAP (opinion)	R-Precision (opinion)	P@10 (opinion)
DUT08_Baseline_Run1	title field	0.3672	0.4264	0.6827	0.2689	0.3318	0.5007
DUT08_Baseline_Run2	expanded	0.3656	0.4241	0.6773	0.2686	0.3320	0.5007

Table 1 Baseline Adhoc Results

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
NOV 2008		**00-00-2008 to 00-00-2008**

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
DUTIR at TREC 2008 Blog Track	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Dalian University of Technology,Information Retrieval Lab,China,**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES
Seventeenth Text REtrieval Conference (TREC 2008) held in Gaithersburg, Maryland, November 18-21, 2008. The conference was co-sponsored bythe National Institute of Standards and Technology (NIST) the Defense Advanced Research Projects Agency (DARPA) and the Advanced Research and Development Activity (ARDA).

14. ABSTRACT
see report

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			
unclassified	**unclassified**	**unclassified**	**Same as Report (SAR)**	**7**	

3. OPINION RETRIEVAL TASK

In Opinion Retrieval Task, the main retrieval unit is permalink. Our method is composed of three steps: Data Preprocessing, Indexing, and Query Construction. The method in [1] was used for data preprocessing. In this chapter, two retrieval steps: topic retrieval and opinionative document ranking are introduced.

3.1 TOPIC RETRIEVAL

Our topic retrieval method mainly includes three phases: concept identification, query expansion, and pseudo-relevance feedback based retrieval.

3.1.1 Concept Identification

In topic retrieval the concepts were identified first. The concept in a query is defined as the single words or phrases that denote an entity. A query may contain multiple concepts. For example, a query "UN Commission on Human Rights" contains three concepts "UN Commission", "Human Rights" and the query "UN Commission on Human Rights" itself. The synonyms of the concepts were also identified.

3.1.2 Query Expansion

Query expansion aims to add a certain number of query-relevant terms to the original query in order to improve retrieval effectiveness. Be different from the general query expansion, here the recapitulative concepts were more focused on. For example, the concept "Frank Gehry architecture" will be expanded as "Walt Disney Concert Hall", "Weisman Art Museum" and other architectures which are designed by Frank Gehry. The online dictionary Wikipedia [2] was utilized to accomplish the expansion.

3.1.3 Pseudo-relevance Feedback Based Retrieval

Lemur [3] was used to carry the pseudo-relevance feedback based retrieval. There are seven retrieval models in Lemur and Indri model was finally selected to retrieval.

3.2 OPINIONATIVE DOCUMENT RANKING

The topic relevant documents have been identified. But actually related to the queries and contain opinions are still need to be found. This is done by using the text window method. First the subjective sentences were found. If a sentence contains at least one sentimental word, it is identified as a subjective sentence. When there is a subjective sentence, two sentences were got to prior to it and two sentences following it to form a 5-sentence window. And then the search was done with the original query terms and the expanded query terms within this window. If certain restrictions are met, this subjective sentence is labeled as a relevant opinionative sentence (ROS). Otherwise it is discarded. A document having at least one ROS is considered to be a relevant opinionative document (ROD) of the query topic. Then the query-document opinion similarity, tagged as $Sim_{op}(d, Q)$, can be calculated by,

$$Sim_{op}(d,Q) = \log(|ROS(d,Q)|+1) \quad (1)$$

where Q denotes a query; d is a document in the ROD set of Q; |ROS(d, Q)| denote the size of ROS set in d for the query Q. The final ranking score (Sim) can be calculated by the following formula:

$$Sim = k * Sim_{ir} + (1-k) * Sim_{op} \quad (2)$$

In the formula above, Sim_{ir} is the query-document opinion similarity obtained by topic retrieval, Sim_{op} is the query-document opinion similarity defined by formula 1.

3.3 RESULTS

Table 2 is the runs and results submitted in Opinion Retrieval Task.

RunID	Judging order	Description	MAP	R-Precision	P@10
DUTIR08Run1	3	topicWords + opinionWords	0.2866	0.3470	0.5193
DUTIR08Run2	4	topicWords + 5Windows	0.3291	0.3768	0.6007
DUTIR08Run3	2	Pseudo Relevance Feedback	0.2964	0.3493	0.5207
DUTIR08Run4	1	Pseudo Relevance Feedback+5Windows	0.3394	0.3818	0.6173

Table 2 The results of Opinion Retrieval Runs Ranked by Opinion

RunID	Judging order	Description	MAP	R-Precision	P@10
DUTIR08Run1	3	topicWords + opinionWords	0.3736	0.4369	0.6980
DUTIR08Run2	4	topicWords + 5Windows	0.4161	0.4604	0.7707
DUTIR08Run3	2	Pseudo Relevance Feedback	0.3925	0.4421	0.7033
DUTIR08Run4	1	Pseudo Relevance Feedback+5Windows	0.4239	0.4674	0.7773

Table 3 The results of Opinion Retrieval Runs Ranked by Topicrel

According to Table 2 and 3, 5 Windows method works better than the method with opinion words method in both the opinion and topic relevance result. On the basis of 5 Windows method, the Pseudo Relevance Feedback contributes to a little improvement on both the opinion and topic relevance result. Moreover, the consistence between the result of Table 2 and Table 3 verifies the necessity of topic relevance retrieval before opinion retrieval.

Best Baseline	Baseline MAP	Best Non-Baseline	Non Baseline MAP	Increase
DUT08BRun1	0.2689	DUTIR08Run4	0.3394	26.22%

Table 4 Improvement over the baselines

According to Table 4, the combination of 5 Windows method and Pseudo Relevance Feedback behaves well in opinion finding task.

4. POLARITY TASK

Polarity classification task last year is developed in this year's Blog Track. The polarity task asks a system to retrieve both the subjective and the negative documents about every topic, and rank the results like in a user, with the mixed opinions not being included. For this task we applied two different approaches. One is a classification approach using SVM, the other is a retrieval method. The queries used in the retrieval method are composed for positive and negative opinions respectively.

4.1 POLARITY IDENTIFICATION AND RETRIEVAL

4.1.1 Classification Method

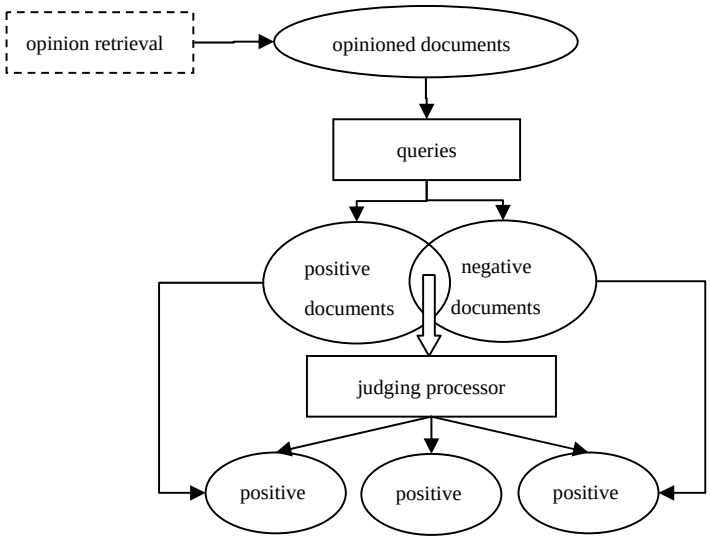
The classification method we used in the polarity task is based on the approach we used in the polarity subtask of Blog Track 2007[1]. We trained two SVM classifiers using the provided results for the 50 topics in Blog Track 2006 as training data. And then the classifiers were used to classify the processed documents.

First, polarity task deals with the relevant documents judged by the opinion retrieval task; these documents were extracted and processed. Then we suppose that not all sentences in the documents are opinioned and relevant. So only the sentences with the extended topic words, which were obtained in the opinion retrieval task, were extracted. We used a two-phase classifier strategy. The first classifier classifies the documents into two categories - relevant-not-opinioned (tagged as *[1]*) and relevant-opinioned (tagged as *[5]*). The second classifier classifies the documents in category *[5]* into three categories, which are relevant-positive-opinions (tagged as *[4]*), relevant-mixed-opinions (tagged as *[3]*), and relevant-negative-opinions (tagged as *[2]*). Documents in categories *[4]* and *[2]* are then the required documents.

4.1.2 Retrieval Method

The retrieval method aims to compose two queries to retrieve all the positive documents and all the negative documents respectively. And then the two retrieved document sets will be compared. Documents in both sets will be judged by a judging processor, which decides weather the document should be tagged as positive (*[4]*), negative (*[2]*), or mixed (*[3]*). Figure 1 below shows the structure of the retrieval method system

Figure 1 The structure of the retrieval method system.



The queries were composed with sentimental lexicons and their weights. The sentimental lexicons used are the same lexicon we used in Blog Track 2007.Their weights are computed with reference from SentiWordNet [4]. The positive documents and negative documents are retrieved with different queries composed respectively by positive sentimental words and negative sentimental words. Indices of the query were built only with documents retrieved in opinion retrieval task. The judging processor is based on the rules below:

$$Polarity(d) = \begin{cases} positive, (P_d - N_d)/T_n > 20\% \\ mixed, others \\ negative, (N_d - P_d)/T_n > 20\% \end{cases} \quad (3)$$

In which P_d is rank of document d in the positive results, N_d is the rank of document d in the negative results, and T_n is total number of results retrieved.

4.3 RESULTS

Table 5 and 6 show the runs and results submitted in polarity task.

RunID	Based on	MAP	R-PREC	P@10
DUTIR08Run2P	DUTIR08Run2	0.0679	0.1185	0.2101
DUTIR08Run5P	DUTIR08Run4	0.0085	0.0282	0.0611
DUTIR08Run6P	DUTIR08Run2	0.0057	0.0261	0.0523

Table 5 Polarity Runs Positive

RunID	Based on	MAP	R-PREC	P@10
DUTIR08Run2P	DUTIR08Run2	0.0515	0.0843	0.1620
DUTIR08Run5P	DUTIR08Run4	0.0086	0.0264	0.0500
DUTIR08Run6P	DUTIR08Run2	0.0071	0.0234	0.0542

Table 6 Polarity Runs Negative

DUTIR08Run2P is the result of the classification method; DUTIR08Run5P and DUTIR08Run6P are the results of the retrieval method. The results suggest that our retrieval method is not as effective as the classification method, and with the same method results based on DUTIR08Run4P is relatively superior to results based on DUTIR08Run2P.

5. BLOG DISTILLATION TASK

In this task, feed files are preferred as retrieval units due to the need to submit feed no. Like the opinion retrieval task, our method is divided into three steps: Data Preprocessing, Indexing, and Query Construction.

5.1 DATA PREPROCESSING

Some manual work such as analyzing the possible formats of the feed files and recording the possible tags that contain desired information according different formats are necessary. Finally three types of feed formats were found according their different displaying styles: RSS, RDF, and ATOM. Moreover, some types of feed could be divided into smaller units such as Item or Entry. Whether for the whole feed or the smaller units in it, desired contents in them are often in the fixed tags such as <description>, <content>, <summary> and so on. So feed files with non English ones removed were later parsed according these tags with htmlparser [6] iteratively. When using htmlparser, these tags need to be defined and registered in order to identify these tags and extract contents among them. The size of original feed is 38.8GB and the size of feed after non English ones and undesired contents are removed becomes 17.5GB.

5.2 INDEXING

Preprocessed data are then indexed with Indri Search Engine [3]. Feeds are different from permalinks, there are often redundant feeds among different files and the contents of them are varying or not. So when running query on the index, it needs to remove the redundant feeds and make a little adjustment.

5.3 EXPANDING THE QUERIES

In this step, we use description and narrative field to expand query. As run1 just uses the field of title, there are irrelevant answers in the results returned by the query, so we employ the other field to get rid of irrelevant answers. In run2, we use the title field and employ the pseudo feed back method [5] to get the answer. In run3, we build another query by using title, description, narrative field and so on. In run4, we use the query of run3 and pseudo feed back to get answers. This is an

automatic run with all fields and is got by merging the pseudo feedback results from Permalinks and Feeds respectively.

5.4 RESULTS

RunID	Judging order	Description	MAP	R-Precision	b-Bref	P@10
DUTIR08D Run1	3	title	0.1600	0.2293	0.2058	0.2600
DUTIR08D Run2	4	title	0.1558	0.2250	0.2046	0.2500
DUTIR08D Run3	2	title, description, narrative	0.1581	0.2222	0.1951	0.2760
DUTIR08D Run4	1	title, description, narrative	0.1632	0.2365	0.2063	0.2780

Table 7 Results of Feed Distillation Runs

DUTDRun4 is relatively superior to others. From Table 7, we can see that pseudo feedback is useful to get precise answers. The MAP, R-Precision, b-Bref and P@10 are the highest of the four runs. From the runs3, we can see that the P@10 is higher than that of Runs1 and Runs2, and the pseudo feedback from runs3 is the highest. So we can say that p@10 is important for pseudo feedback. If we can use precise pages for pseudo feedback, we can get much higher results.

REFERENCES

[1] DUTIR at TREC 2007 Blog Track, http://trec.nist.gov/pubs/trec16/papers/dalianu.blog.final.pdf
[2]Wikipedia, http://en.wikipedia.org/wiki/Wikipedia
[3]The lemur Toolkit: http://www.lemurproject.org/
[4] http://sentiwordnet.isti.cnr.it/download_.0/.
[5] Jonathan Elsas, Jaime Arguello, Jamie Callan, Jaime Carbonell, Retrieval and Feedback Models for Blog Distillation, http://www.cs.cmu.edu/~jaime/TREC07ElsasJ.pdf
[6] HTML Parser Tool Kit: http://sourceforge.net/project/showfiles.php?group_id=24399